≢⊏ Fermilab

New Features in TRANSPORT

David C. Carey

September 7, 1981

The computer program TRANSPORT, used for designing charged particle beam transport systems, has been modified to include a number of new features. Among them are the possibility of using accelerator notation to specify the beam matrix, expanded fitting capabilities including the ability to constrain algebraic combinations of matrix elements, and more flexible means of specifying individual or groups of beam line elements.

This document is to be used as a supplement to the TRANSPORT manual. It is organized by type code, as is the manual. The contents of this report will eventually be written into a revised TRANSPORT manual. In a few instances, the material of the manual has been rewritten, but in most cases it has simply been expanded.

INPUT BEAM: Type code 1.0

The beam matrix may also be specified in terms of accelerator notation using the parameters β, α, and η. A 13. 7. ; element placed before the beam element will indicate that the beam matrix input and output and beam matrix constraints are all expressed in accelerator notation. The meaning of the accelerator parameters β and α is shown in the illustration below. The parameters β and α may be defined for each transverse plane. The parameter η is used to characterize the behavior of off-momentum trajectories. The use of this option is not compatible with either misalignments (type code 8.) or second order (type code 17.).

If accelerator notation is to be used, there are eight entries to be made on the beam card.

1 - The type code 1.0 (specifies a beam entry follows).

2 - β_x

3 - α_x

4 - β_y

5 - α_y

6 - 0.

7 - 0.

8 - The momentum of the central trajectory $\left[p(0) \right]$ (GeV/c).

The units for β are the quotient of the transverse length units and the transverse angle units. Conventionally β is expressed in mm/mr or meters. Following this convention requires a change in the transverse distance unit to mm using the 15. type code. The six components of the η vector are specified

using the centroid shift (type code 7.). If no type code 7
element is present, the fractional momentum deviation of η
from the beam design momentum is one unit in fractional momentum
spread. Normally this is in percent $\Delta p/p$, but can also be
changed via a 15. type code entry. All other coordinates of
η are then taken to equal zero. A beam correlation entry (type
code 12.) is not used with accelerator notation since α_x and
α_y express the degree of correlation between position and angle
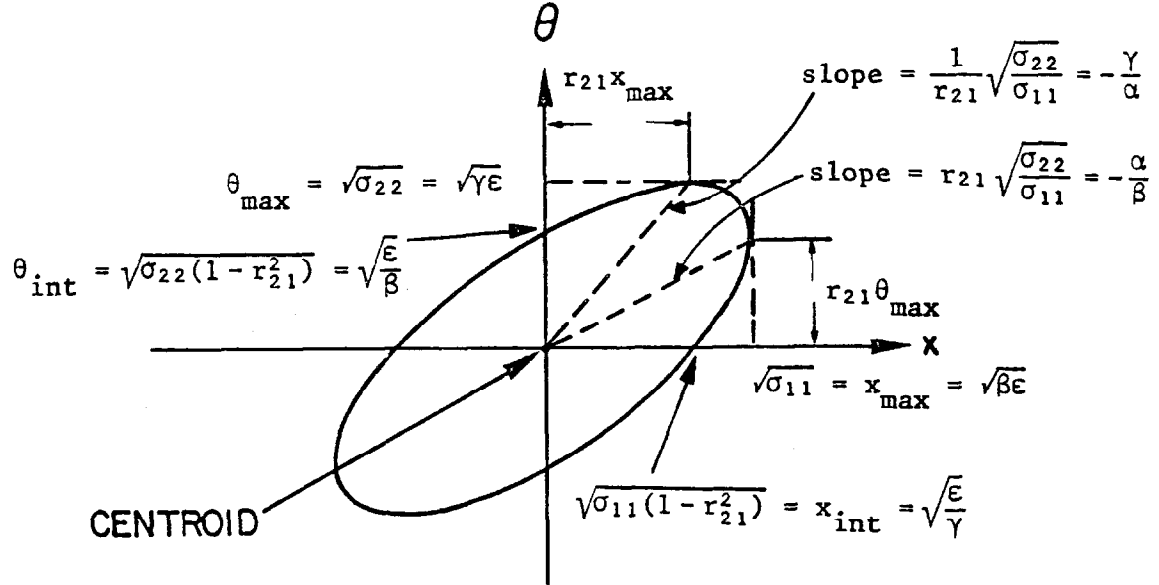in the two transverse planes.

The beam matrix, whenever printed, will be expressed in
accelerator notation. If the beam matrix is printed after every
physical element, by the insertion of a 13. 3. ; element, a
heading will be printed to identify the numbers in the beam
matrix. The beam matrix will be printed in a single line. The
quantities printed are

$$\psi_x \quad \psi_y \quad \beta_x \quad \beta_y \quad \alpha_x \quad \alpha_y \quad \eta_x \quad \eta_y \quad \eta_x' \quad \eta_y'$$

The parameters ψ_x and ψ_y are the phase advances in the horizontal
and vertical planes. The phase advance is given by

$$\psi = \int \frac{ds}{\beta}$$

where s is the distance along the reference trajectory. The units
for ψ are degrees. If the beam matrix is printed only after
selected elements via a 13. 1. ; type code entry, no heading
is printed.

The figure contains the following labels:

θ

$r_{21}x_{max}$

$\text{slope} = \dfrac{1}{r_{21}}\sqrt{\dfrac{\sigma_{22}}{\sigma_{11}}} = -\dfrac{\gamma}{\alpha}$

$\text{slope} = r_{21}\sqrt{\dfrac{\sigma_{22}}{\sigma_{11}}} = -\dfrac{\alpha}{\beta}$

$\theta_{max} = \sqrt{\sigma_{22}} = \sqrt{\gamma\epsilon}$

$\theta_{int} = \sqrt{\sigma_{22}(1-r_{21}^2)} = \sqrt{\dfrac{\epsilon}{\beta}}$

$r_{21}\theta_{max}$

X

$\sqrt{\sigma_{11}} = x_{max} = \sqrt{\beta\epsilon}$

$\sqrt{\sigma_{11}(1-r_{21}^2)} = x_{int} = \sqrt{\dfrac{\epsilon}{\gamma}}$

CENTROID

A TWO-DIMENSIONAL BEAM PHASE ELLIPSE

The area of the ellipse is given by:

$$A = \pi(\det \sigma)^{\frac{1}{2}} = \pi x_{max}\,\theta_{int} = \pi x_{int}\,\theta_{max}$$

The equation of the ellipse is:

$$\gamma x^2 + 2\alpha x\theta + \beta\theta^2 = \epsilon$$

where

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{21} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \epsilon \begin{bmatrix} \beta & -\alpha \\ -\alpha & \gamma \end{bmatrix}$$

and

$$\beta\gamma - \alpha^2 = 1 \;, \quad r_{21} = r_{12} = \frac{-\alpha}{\sqrt{1+\alpha^2}} = \frac{-\alpha}{\sqrt{\beta\gamma}}$$

FRINGING FIELDS and POLE-FACE ROTATIONS for bending magnets
Type code 2.0

If the bending magnets to be specified are rectangular
as seen from the top, then the sum of the input and output
pole face rotation angles will equal the bend angle. Several
options exist for automatically setting the pole face rotation
angles in terms of the bend angle. Such options are given by
entries of 13. 40. ; through 13. 43. ; inclusive. Even
when one of these options is selected, the 2. element must
be included in the data before and after the type code 4. entry
if fringing-field effects are to be calculated. Details of
the various options are found in the section describing type
code 13.

WEDGE BENDING MAGNET: Type code 4.0

Alternatively, a wedge bending magnet may be specified in terms of its length and bend angle. The four parameters to be specified are then

1 - Type code 4.0 (specifying a wedge bending magnet)

2 - The (effective) length L of the central trajectory in meters.

3 - The bend angle α in degrees.

4 - The field gradient (n-value).

The quantities L, α, and n may be varied. The units for bend angle may be changed via a 15. 7. element preceding the beam card.

If a 13. 48. ; entry is inserted in the data, all subsequent bend magnet entries are taken to be expressed in terms of L, α, and n. At a later point one can revert to the normal description in terms of L, B, and n by the insertion of a 13. 47. ; element.

QUADRUPOLE: Type code 5.0

A device which is focusing in both transverse planes or defocusing in both transverse planes may also be represented via a 5. type code. The description is the same as for an ordinary quadrupole, except that the sign of B now applies to both planes. A 13. 98. ; element inserted before a 5. type code will cause it to be taken as such a device. A subsequent 13. 97. ; entry causes the 5. type code to be taken as representing a quadrupole.

SHIFT IN THE BEAM CENTROID: Type code 7.0

If accelerator notation is selected for the beam matrix
via a 13. 7. ; element, then the 7. type code is used to
represent the η function. The coordinates specified are
those of the initial η function. The η function is conventionally
taken to represent a trajectory whose momentum differs by one
unit from the design momentum of the beam line. In this case
the δ parameter on the shift element would be 1.0.

VARY CODES and FITTING CONSTRAINTS: Type code 10.0

First-order vary codes

In a first-order run, the additional quantities marked v may be varied

BEAM (rotated ellipse).. 12.vvvvvvvvvvvvvvv - All elements of

                    the correlation matrix may be varied.

Second-order vary codes

In a second-order run the following parameters may be varied:

ROTAT.......... 2.v - The pole face angle of a bending magnet

                    may be varied.

DRIFT.......... 3.v - The drift length may be varied.

BEND........... 4.vvv - The length, the field, and/or the n-value

                    may be varied. Alternatively, the length, the

                    bend angle, and/or the n-value may be varied.

QUAD........... 5.vv0 - The length may be varied; the field may

                    be; the aperture may not be.

MATRIX......... 14.vvvvvv0 - Any of the first order matrix elements

                    may be varied.

INITIAL   16.0v - Any of the three initial position floor
COORDINATES....

                    coordinates or two angle coordinates may be varied.

$\varepsilon(1)$........... 16.0v 1. - The normalized quadratic term (sextupole

                    component) in the midplane expansion for the field

                    of a bending magnet may be varied.

1/R1........... 16.0v 12. - The pole face curvature of a bending

                    magnet entrance may be varied.

1/R2........... 16.0v 13. - The pole face curvature of a bending

magnet exit may be varied.

SEXTUPOLE...... 18.0v - The field strength may be varied.

SOLENOID....... 19.vv - The length and/or field may be varied.

BEAM ROTATION.. 20.v - The angle of rotation may be varied.

Variation of any parameter which may also be varied in first order should be done with caution as it may affect the first-order properties of the beam line. But, inversely coupled drift spaces straddling a sextupole will, for example, show only second-order effects.

The special parameter cards (type code 16.0) once introduces apply to all subsequent magnets in a beam line until another type code 16.0 specifying the same parameter is encountered. Thus, if such a parameter is varied, the variation will apply simultaneously to all subsequent magnets to which it pertains. The variation will persist until the parameter or vary code attached to the parameter is changed by the introduction of another type code 16.0 element specifying the same parameter.

The first-order parameters of a pole face rotation, bend, or solenoid may not be fit to satisfy a second-order constraint. They are variable in a second-order run only to permit first- and second-order constraints to be imposed simultaneously.

Possible fitting constraints

Any elements of the first-order transfer matrix, the floor coordinates, and the system length may now also be fit in a second-order run.

Beam matrix fitting constraints using accelerator notation

The correspondence between the notation for a beam constraint and the convention for beam parameter input is the same as when accelerator notation is not used. There are five parameters to be specified.

1 - Type code 10.

2 - Code digit (i).

3 - Code digit (j).

4 - Desired value of parameter to be fit.

5 - Desired accuracy of fit (standard deviation).

If $i = j$, the constraint applies to the i'th physical parameter on the beam(type code 1.) element. For example, if β_x is to be constrained, then $i = j = 1$. If $i = 0$, the constraint applies to the j'th physical parameter on the shift (type code 7.) element. Thus, to fit η_x, one would set j equal to 1.

Some typical accelerator notation beam constraints are as follows:

Desired optical condition	Typical fitting constraint
$\beta_x = .5$	10. 1. 1. .5 0.001 'F1' ;
$\alpha_x = 0$	10. 2. 2. 0. 0.001 'F2' ;
$\beta_y = .3$	10. 3. 3. .3 0.001 'F3' ;
$\alpha_y = -.2$	10. 4. 4. -.2 0.001 'F4' ;
$\eta_x = .5$	10. 0. 1. .5 0.001 'F5' ;
$\eta'_y = -.2$	10. 0. 4. -.2 0.001 'F6' ;

Algebraic combination of matrix elements constraint

Any quantity which can be constrained may also be used to
form an algebraic expression which may itself be constrained.
The algebraic combinations are formed using the 22. and 23.
type code elements. The results are placed in numbered registers.
To constrain the contents of a numbered register, five parameters
are required.

1 - Type code 10.

2 - Code digit 9.

3 - Register number (j).

4 - Desired value of register contents.

5 - Desired accuracy of fit.

Complete descriptions on the formation of algebraic
combinations of matrix elements and their placement in storage
registers are given in the sections on type codes 22. and 23.

The following example shows the use of the 22, 23, and 10
type code elements. It shows a fit of $R_{11} + R_{22}$ to a value of .2.

```
22.  -1.  1.  1.    ;
22.  -2.  2.  2.    ;
23.   1.  2.  1.  1.  ;
10.   9.  1.  .2  0.001  ;
```

INPUT-OUTPUT options: Type code 13.

Several new options for input-output control have been implemented. The format for type code 13 is:

1 - Type code 13.0

2 - Code number

The new options with their code numbers are as follows;

Accelerator notation for beam matrix 7.

Both input and output for the beam matrix are in accelerator notation. The parameters β, α, and η_x of accelerator theory are used. Constraints on the beam are also taken to be in terms of β, α, and η. The forms of input and output for the beam matrix are explained under type codes 1 and 7. The form of constraints on the beam are explained under type code 10. To specify that accelerator notation is to be used in a run, the 13. 7. ; element must be before the beam element (type code 1.).

Positions of waists 14.

The longitudinal positions and transverse dimensions of (real or virtual) waists in both transverse planes are printed at the point of insertion of this element. The waist location and characteristics are computed from the dimensions of the current location assuming infinitely long drift spaces both upstream and downstream.

Inverse transfer matrix print control -4.

The inverse of the current transformation matrix R1 will

be printed by this code. If the program is computing a second-order matrix, the inverse of the second-order transformation matrix will be included in the print out.

Refer transfer matrix to original coordinate system 13.

If the transverse coordinates are rotated through an angle via a 20. type code entry, the rotation is taken only as specifying the orientation of the rotated magnet. The transfer matrix, whenever printed, is given in the unrotated coordinate system. The 13. 13. ; element must precede any 20. type code elements to be used. When using a 13. 13. ; element, all coordinate rotations must be (positive or negative) multiples of 90°.

Shift in reference trajectory 9.

At the location of this element, the reference trajectory is shifted to line up with the first-order image of the original beam centroid. Thus, the reference trajectory of the beam line, as followed by the program will not be continuous. Quadrupoles will also show dispersive effects. All the fitting options connected with this element operate as expected. Thus the original beam centroid displacement parameters can be varied to fit values of floor coordinates specified after the shift.

Precise values of varied parameters 16.

The values of any varied parameters will be printed in F18.10 format. This option is useful primarily for investigating the mathematical characteristics of a solution.

Pole face rotation angle specification 40., 41., 42., 43.

The pole face rotation angle normally specified with type code 2. may alternatively be calculated from the bend angle of the associated bend magnet. The pole face rotation element must still be present. The options described here merely cause the value of β (the pole face rotation angle) to be filled in automatically by the program.

The element specifying the means of determination of the pole face rotation angle must precede the bending magnet specification, including pole face rotation angle elements, to which it applies. It will remain in effect until the option is respecified via another 13. type code element. Some caution must be taken in the use of these options. If a bend magnet is segmented, then the pole face rotation angle will be calculated from the bend angle of the adjacent segment.

(13. 41. ;) Both input and output pole face rotation angles are equal to half the bend angle.

(13. 42. ;) The entry pole face rotation angle will be zero. The exit pole face rotation angle will equal the bend angle.

(13. 43. ;) The entry pole face rotation angle will equal the bend angle. The exit pole face rotation angle will be zero.

(13. 40. ;) The normal option is restored. The pole face rotation angles will be read from the data.

Bend magnet input specifications 47., 48.

(13. 48. ;) Bending magnets encountered subsequent to the insertion of this element are to be specified by the length, bend angle, and n value. Thus a bend magnet element will take the form

4. L α n ;

(13. 47. ;) The normal option for specification of a bending magnet is restored by this element. A bend magnet element now takes the form

4. L B n ;

Lithium or plasma lens 97., 98.

(13. 98. ;) All quadrupoles subsequent to this element will be taken as having the same effect in both transverse planes. Thus, if the pole tip field is positive, both planes will be focusing. If the pole tip field is negative, both planes will be defocusing.

(13. 97. ;) The normal option is restored for a quadrupole element. It is now focusing in one plane and defocusing in the other.

SPECIAL INPUT PARAMETERS: Type code 16.0

Code digit for new special parameter

2. K_O – An integral related to the transverse displacement

of the reference trajectory arising from passage

through the fringe field of a bending magnet. If

the (16. 5. g/2 ;) element has been inserted,

the program assumes a default value of $K_O = .5$.

Insertion of the 16. 2. K_O element supplants this

default value. The transverse displacement is given

by $\Delta x = g^2 K_O \rho$. The actual value of K_O is given by

a double integral

$$K_O = \frac{1}{g^2 B_O \cos^2 \beta} \int_{-s_1}^{\infty} ds \int_{-s_1}^{\infty} (B_O - B(s)) \, ds$$

where s is measured perpendicular to the pole face.

The parameter β is the pole face rotation angle and

B_O is the field interior to the magnet. The value

of s_1 is chosen so as to be well within the interior

of the magnet.

COORDINATE ROTATION: Type code 20.0

Normally, the transformation matrix is expressed in the rotated system. If a 13. 13. element is included in the data before the beam (type code 1.) element, the transformation matrix will be expressed in the original (unrotated) coordinate system. In this latter case, all coordinate rotations must be (positive or negative) multiples of 90°.

ALGEBRAIC COMBINATIONS OF MATRIX ELEMENTS - DEFINING REGISTER CONTENTS: Type code 22.

Any of the quantities which can be constrained can also be used to form algebraic combinations which in turn may be constrained. A set of ten storage registers is available for formation of such algebraic combinations. The 22. type code element is used for storage into these registers. Four parameters are required.

1 - Type code 22. (indicating a storage operation)

2 - Code digit i.

3 - Code digit j.

4 - Register number

The indices i and j have the same meaning as for the fitting constraint (10.) type code. Thus, for example, if one wished to place the R_{34} matrix element into register 1, one would use the following element

22. -3. 4. 1. ;

Numerical constants may also be placed in registers for use in forming algebraic expressions. If i is equal to 100, then the third parameter is taken to be the numerical constant. For example, the element

22. 100. 3.14159 7. ;

causes the number 3.14159 to be placed in register 7.

The contents of a register may be constrained by a fitting constraint element (type code 10.). Details are given in the section describing fitting constraints.

ALGEBRAIC COMBINATIONS OF MATRIX ELEMENTS -
FORMING COMBINATIONS: Type code 23.

Algebraic combinations of elements in storage registers may be formed and themselves placed in storage registers. The combinations formed may be constrained or used to form further algebraic combinations.

Five parameters are required:

1 - Type code 23. (signifying an algebraic combination)

2 - First input storage register

3 - Second input storage register

4 - Algebraic operation to be performed

5 - Output storage register

The algebraic operations are referred to by number according to the following list

1 - addition (+)

2 - subtraction (-)

3 - multiplication .(×)

4 - division (÷)

5 - square root ($\sqrt{}$)

If the square root is selected, only the first input register is used. A dummy index should be inserted for the second input register.

As an example of the use of the algebraic combination element we illustrate the process of taking the quotient of the contents of registers 2 and 7 and placing the result in register 2

23.   2.   7.   4.   2.

DEFINED SECTION: Type code 24.

A system may contain a section which is repeated at some later point. The repeated section may not occur immediately after the original section, so that the repeat element (type code 9.) is not appropriate. It may also be useful to repeat a section, but with the elements listed in the reverse order.

The 24. type code element allows definition of the section to be repeated. It also indicates the locations at which the section is to be repeated and whether the repetition is to be forwards or backwards.

Three parameters are required:

1 - <u>Type code 24.</u>

2 - Code digit.

3 - Section name. The section name is a maximum of four characters long and is enclosed in single quotes.

The meaning of the code digit is as follows:

1 - Indicates the beginning of the section to be defined.

2 - Indicates the end of the section to be defined.

3 - The section named is to be repeated at the present location in the forwards direction.

4 - The section named is to be repeated at the present location with the elements in reverse order

A number of rules apply to the use of the defined section element. Each defined section must have both its beginning and its end indicated, and the beginning must precede the end. The

definition of a section must completely precede its use. A
given name can be used only once to define a section, although
it can be used many times to indicate a repeat of a section.
A defined section may not refer to itself, either explicitly
or implicitly. The number of defined sections is limited to
10. The total z rotation (type code 20.) within a defined
section must sum to zero. Finally, defined sections must
nest properly with the repeat code (type code 9.). Defined
sections, however, need not nest properly with each other.

Example of a defined section

24. 1. 'DOUB' ; ——————— Begin defined section

5. 10. 5. 2. ;

3. 8. ;

5. 10. -5. 2. ;

24. 2. 'DOUB' ; ——————— End defined section

24. 3. 'DOUB' ; ——————— Repeat defined section

24. 4. 'DOUB' ; ——————— Repeat defined section in
                                              reverse order